Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 90542, Pages 1–13
DOI 10.1155/ASP/2006/90542

Least-Square Prediction for Backward Adaptive Video Coding

Xin Li

Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV 26506, USA

Received 27 July 2005; Revised 7 February 2006; Accepted 26 February 2006

Almost all existing approaches towards video coding exploit the temporal redundancy by block-matching-based motion estimation and compensation. Regardless of its popularity, block matching still reflects an ad hoc understanding of the relationship between motion and intensity uncertainty models. In this paper, we present a novel backward adaptive approach, named "least-square prediction" (LSP), and demonstrate its potential in video coding. Motivated by the duality between edge contour in images and motion trajectory in video, we propose to derive the best prediction of the current frame from its causal past using least-square method. It is demonstrated that LSP is particularly effective for modeling video material with slow motion and can be extended to handle fast motion by temporal warping and forward adaptation. For typical QCIF test sequences, LSP often achieves smaller MSE than 4×4, full-search, quarter-pel block matching algorithm (BMA) without the need of transmitting any overhead.

1. INTRODUCTION

Motion plays a fundamental role in video coding. Motion compensated prediction (MCP) [1] represents the most popular approach towards exploiting the temporal redundancy in video signals. In hybrid MCP coding [2], a motion vector (MV) field is estimated and transmitted to the decoder and motion compensation (MC) is the key element in removing temporal redundancy. In the past decades, constant progress has been made to an improved understanding of the relationship between motion and intensity uncertainty models under the framework of hybrid MCP coding, which culminated in the latest H.264/AVC video coding standard [3, 4].

Despite the triumph of hybrid MCP coders, MC only represents one class of solution to exploit the temporal redundancy. The apparent advantage of MC is its conceptual simplicity—the optimal MV that most effectively resolves the intensity uncertainty is explicitly transmitted to the decoder. To keep the overhead not to outweigh the advantages of MC, a coarse MV field (block-based or region-based) is often used. The less obvious disadvantage of MC is its (over)commitment to motion representation. Such commitment is particularly questionable as the motion gets complex. Take an extreme example—in the case of nonrigid motion, it often becomes more difficult to justify the benefit of MC.

In this paper, we present a new paradigm for the video coding that does not *explicitly* perform motion estimation (ME) or MC. Instead, temporal redundancy is exploited by a backward adaptive spatiotemporal predictor that attempts to make the best guess of the next frame based on the causal past. The support of temporal prediction neighbors is updated on-the-fly in order to cover the probability distribution function (pdf) of MV field (note that we do not need to estimate any motion vector but only its distribution for any frame). Motivated by a duality between geometric constraint of edges in still images and iso-intensity constraint along motion trajectory in video, we propose to locally adapt the predictor coefficients by least-square (LS) method, which is given the name "least square prediction" (LSP).

A tantalizing issue arising from such backward adaptation is its capability of modeling video source. An ad hoc classification of video source based on motion characteristics is shown in Figure 1. The primary objective of this paper is to demonstrate that LSP is particularly suitable for modeling the class of slow and natural motion regardless of the motion rigidity. Slowness is a relative concept—at the frame rate of 30 fps, we assume that the projected displacement of any physical point in the scene due to camera or object motion is reasonably small (e.g., fewer than 10 pixels). Naturalness refers to the acquisition environment—natural scene, normal lighting, stabilized camera, and no post-production editing (e.g., artificial wipe effect).

It is from such modeling viewpoint that we argue that LSP has several advantages over hybrid MCP. First, backward adaptive LSP does not suffer from the limitation of explicitly representing motion information in forward adaptive approaches. Such freedom from approximating the true motion field leads to more observable coding gain as motion

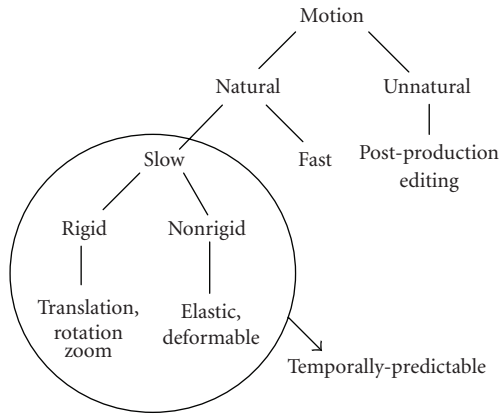


Figure 1: Ad hoc classification of motion in video sequences: we target at the modeling of slow and natural motion that is temporally predictable.

gets more complex but remains temporally predictable (e.g., camera zoom). Second, LSP inherently attempts to find the best tradeoff between spatial and temporal redundancies to resolve intensity uncertainty, which is desirable in handling the situations such as occlusions. Last but not the least, it is possible to extend LSP by temporal warping and forward adaptation to handle certain type of video with fast or disturbed motion, which improves the modeling capability.

Experimental results with a wide range of test sequences are very encouraging. Without transmitting any overhead, LSP can achieve even better accuracy than 4×4, full-search, quarter-pel block matching algorithm (BMA) for typical slow-motion sequences. We note that BMA with such setting represents the current state-of-the-art in hybrid MCP coding (e.g., H.264 standard [4]). The prediction gain is particularly impressive for the class of temporally predictable events (motion trajectory is locally smooth within a spatiotemporal neighborhood). The chief disadvantage of backward adaptive LSP is the increased decoding complexity because decoder also needs to perform LSP.

The rest of this paper is organized as follows. Section 2 revisits the role of motion in video coding and emphasizes the difference between forward and backward adaptive modeling. Section 3 deals with the basic formulation of LSP and covers theoretical interpretation based on the 2D-3D duality. Section 4 presents the backward adaptive update of LSP support and analyzes the spatiotemporal adaptation. Section 5 introduces temporal warping to compensate camera panning and forward adaptive selection of LSP parameters. In Section 6, we use extensive experimental results to compare the prediction efficiency of both LSP and BMA. We make some final concluding remarks in Section 7.

2. ROLE OF MOTION REVISITED IN VIDEO CODING

2.1. Bless and curse of motion in video coding

Video source is more difficult to model than image source due to the new dimension of time. In the continuous space, temporal redundancy is primarily characterized by motion, namely, intensity values along the motion trajectory remain constant assuming invariant illumination conditions. However, there exists a fundamental conflict between the continuous nature of motion and discrete sampling of video signals, which makes the exploitation of temporal redundancy difficult. Even a small (subpixel) deviation of the estimated MVs from their true values could give rise to significant prediction errors for spatially-high-frequency components (e.g., edges or textures).

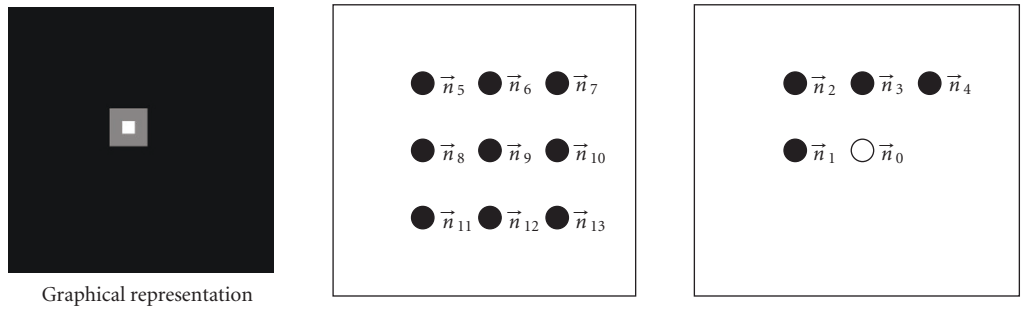
The task of exploiting motion-related temporal redundancy is further complicated by the diversity of motion models in video. Even if for the class of video with rigid motion only (translation, rotation, zoom), ME is twisted with motion segmentation problem [5] when the scene consists of multiple objects at the varying depth. Despite the promise of object-based (region-based) video coding [6], its success remains uncertain due to the difficulty with motion segmentation (one of the long-standing open problems in computer vision). For the class of nonrigid motion, the benefit of MC becomes even harder to justify. For example, the isointensity assumption often does not hold due to the geometric deformation (e.g., flowing fluid) and photometric variation.

Those observations suggest that video coders wisely exploit motion-related temporal redundancy to resolve the intensity uncertainty. Since motion field is both spatially and temporally varying, video source is a nonstationary process. However, when projected to a low-dimensional subspace (e.g., within an arbitrarily small space-time cube), video is locally stationary. Classification is an effective tool for handling such nonstationary sources as image and video. The interplay between classification and rate-distortion analysis has been well understood for still images (e.g., wavelet-based image coding [7–9]). However, motion classification has not attracted sufficient attention from video coding community so far. We will present a review of existing modeling approaches from the adaptive classification point of view.

2.2. Adaptive modeling of video source

Most existing hybrid MCP coders can be viewed as classifying the video source in a forward adaptive fashion. A video frame is decomposed into nonoverlapping blocks and each block is assigned an optimal motion vector found by searching within the reference frame.More sophisticated forward adaptation involves multiple hypotheses [10] (e.g., long-term memory MC [11], overlapped block MC [12]) and region-based MC (e.g., segmentation-based [13]). The major concern with forward adaptive approaches is that the overhead might outweigh the advantages of MC. Such issue involves both the estimation and representation of motion, which often makes it difficult to analyze the overall coding efficiency of hybrid MCP coders.

By contrast, backward adaption is an attractive alternative in that we do not need to transmit any overhead—decoder and encoder operate in a synchronous mode to predict the current frame based on its causal past. Backward



Graphical representation of temporal neighbors (a)

Temporal neighbors (b)

Spatial neighbors (c)

FIGURE 2: An example of predictor based on 13 spatiotemporal causal neighbors (note that the ordering among them does not matter).

adaptation allows us to afford more flexible motion models than block-based ones to resolve the intensity uncertainty. Existing backward adaptive approaches [14, 15] exploit such advantage by segmenting the motion field into regions instead of blocks. Region-based segmentation is essentially equivalent to the layered representation [16] that decomposes video into multiple motion layers. However, subpixel MC remains difficult to be incorporated into the backward framework because subpixel displacement along the motion trajectory often does not exactly match the sampling lattice of a new frame. Due to the importance of motion accuracy in video coding [17], difficulty with subpixel MC appears to be one of the major obstacles in the development of backward adaptive video coders.

To fully exploit the flexibility offered by backward adaptation, we argue that *explicit* estimation of motion field is neither necessary nor sufficient for exploiting the temporal redundancy at least for the class of slow natural motion. Instead, we advocate an *implicit* approach of MC that does not need to estimate MV at all. In our approach, motion information is embedded into a new representation, namely *prediction coefficient vector field*, which can be shown to achieve implicit yet dense (pixel-wise) and accurate (subpixel) MC. The basic idea behind our approach is that instead of searching the optimal MC in forward adaptive scheme, we propose to locally learn the covariance characteristics within a causal window and use it to guide the spatiotemporal prediction.

3. LEAST-SQUARE PREDICTION: BASIC DERIVATION

As the starting point, we will study the simplified case—video containing little motion. Though such class of video is apparently limited, it is sufficient for our purpose of illustrating the basic procedure of LSP. We will first introduce some notation to facilitate the derivation of the closed-form solution of LSP and then provide a theoretical explanation of how LSP tunes the prediction support along the iso-intensity trajectory in the spatiotemporal domain using the 2D-3D duality.

3.1. Least-square prediction

Suppose $\{X(k_1, k_2, k_3)\}$ is the given video sequence within a shot (no scene change) where $(k_1, k_2) \in [1, H] \times [1, W]$ are the spatial coordinates and k_3 is the temporal axis. For the simplicity of notation, we use vector $\vec{n}_0 = [k_1, k_2, k_3]$ to denote the position of a pixel in space-time and its causal neighbors are labeled by $\vec{n}_i$, $i = 1, 2, \ldots, N$. Figure 2 shows an example including four nearest neighbors in space plus nine closest in time [18] (note that their ordering does not matter because it does not affect the prediction result). Under the little-motion assumption, we know the correspondent of $X(\vec{n}_0)$ in the previous frame is likely to be located within the 3×3 window centered at (k_1, k_2). Therefore, we can formulate the prediction of $X(\vec{n}_0)$ from its spatiotemporal causal neighbors by

$$\hat{X}(\vec{n}_0) = \sum_{i=1}^{N} a_i X(\vec{n}_i), \tag{1}$$

where N is the order of linear predictor (it is thirteen in the example of Figure 2). In contrast to explicit ME, motion information is implicitly embedded in the prediction coefficient vector field $\vec{a} = [a_1, \ldots, a_N]^T$. Note that (1) includes both spatial and temporal causal neighbors, which allows the adaptation between spatial and temporal predictions because $\vec{a}$ is seldom a delta function (we will illustrate such adaptation in Section 4.2).

Under the assumption of Markov property with motion field, the optimal prediction coefficients $\vec{a}$ can be trained from a local causal window in space-time. For example, we might use a 3D cube $C(T_1, T_2) = [-T_1, T_1] \times [-T_1, T_1] \times [-T_2, -1]$ centered at $\vec{n}_0$, which gives rise to the total of $M = (2T_1 + 1)^2 T_2$ samples in the training window. Similar to the 2D case, we can write all training samples into an $M \times 1$ column vector $\vec{y}$. If we put the N causal neighbors for each training sample into a $1 \times N$ row vector, then all training samples generate a data matrix C with size of $M \times N$. The derivation of locally optimal prediction coefficients $\vec{a}$ is formulated into the following least-square problem [19]:

$$\min ||\vec{y}_{M\times 1} - C_{M\times N}\vec{a}_{N\times 1}||^2, \tag{2}$$

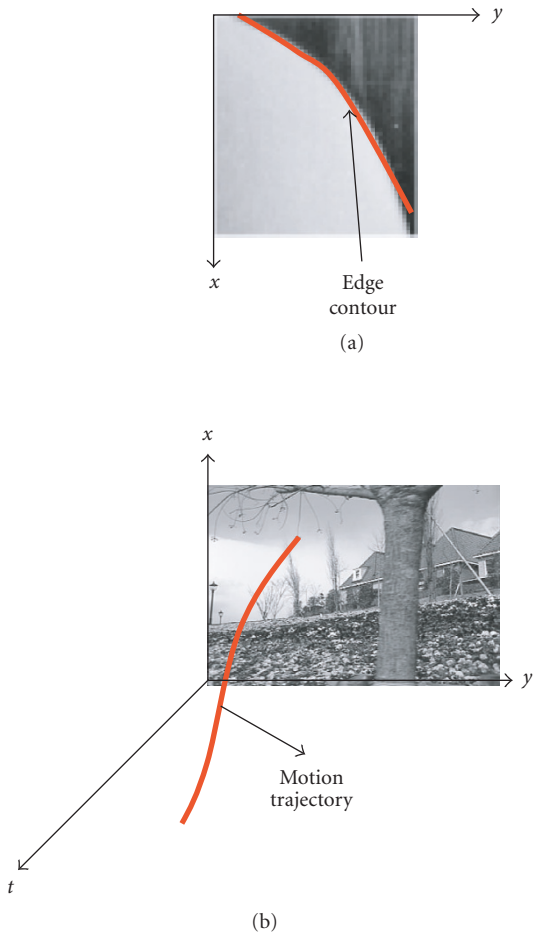


FIGURE 3: Duality between (a) edge contour in still images and (b) motion trajectory in video.

and its closed-form solution is given by

$$\vec{a} = (C^T C)^{-1}(C^T \vec{y}). \tag{3}$$

3.2. Theoretical analysis based on 2D-3D analysis

The suitability of using covariance estimation as an alternative to ME can be best illustrated by the 2D-3D duality, which is introduced next. The duality between 2D image and 3D video can be understood by referring to Figure 3. If we intentionally confuse spatial coordinates with temporal axis, an image consisting of parallel rows (1D signals) is dual to a video consisting of parallel frames (2D signals). Taking the shoulder portion of *lena* image as an example, we can easily observe the following geometric constraint of edge [20]: intensity field is constant along the edge orientation. Therefore, conceptually the contour of an edge in 2D is equivalent to the motion trajectory in 3D—they both characterize the iso-intensity level set in the continuous space. Such duality suggests that mathematical tools useful for exploiting geometric constraint of edges lend themselves to exploiting motion-related temporal redundancy as well.

Specifically, we note that in 2D predictive coding of image signals [21], no estimation of edge orientation is required; instead, the orientation information is learned from the covariance attributes estimated within a local causal window and embedded into a linear predictor whose weights are adjusted on a pixel-by-pixel basis. The support of linear predictor is tuned to match the local geometry regardless of the edge orientation. Using the duality, we might envision a 3D predictive coding scheme without explicit estimation of motion trajectory. Similar to the 2D case, the motion information can be learned from the causal past and embedded into a linear predictor with adjustable weights.

To simplify our analysis of LSP, we opt to drop the vertical coordinate k_2 and consider a slice along the coordinate of (k_1, k_3), as shown in Figure 4. Such strategy essentially reduces the analysis to 2D by only taking the horizontal motion into account.[1] In fact, the concept of spatiotemporal slice is well known in the literature of motion analysis [22, 23] and has found many successful applications from scene change detection to shot classification. Here, we use spatiotemporal slice as a tool for facilitating the analysis of LSP.

Figures 4(a) and 4(b) show the spatiotemporal slices for two popular types of motion: camera panning and camera zoom. The flow-like pattern in those slices corresponds to the motion trajectory along which iso-intensity constraint is satisfied. Intuitively, such pattern can be thought of as geometric constraint of "motion edges." Statistical tools such as LS are known to be suitable for tuning the predictor support to align with an arbitrarily-oriented edge. Therefore, spatiotemporal LSP is also capable of predicting along the motion trajectory as long as local training window contains sufficient relevant data.

It is also enlightening to analyze LSP in the scenario of aperture. Aperture is a problem with explicit motion estimation (e.g., optical flow), which states that the motion information can only be reliably estimated along the normal direction [24]. Such nonuniqueness of solutions calls for regularization in ME (e.g., smoothness constraint in Horn-Schunck method [25]). When local spatial gradients are not sufficient to resolve the ambiguity of MVs along the tangent direction, the rank of the covariance matrix $(C^T C)$ is not full, which implies that multiple MMSE solutions exist. However, since we do not need to distinguish them (i.e., multiple MMSE predictors work equally well on resolving the intensity ambiguity of the current pixel), aperture does not cause any difficulty to LSP.

As we consider more general motion such as camera rotation or zoom, motion trajectory of an object becomes more complicated curves in 3D (e.g., spirals, rays). However, locally within a small spatiotemporal cube, the flow directions of motion trajectory is still approximately constant. Therefore, LS-based adaptation is still able to tune the predictor support to match the dominating direction within the local training window. As the training window moves in space and time, the dominating direction slowly evolves, so

[1] Nevertheless, horizontal motion is often more dominant than vertical motion in typical video sequences.

(a) panning (b) zoom (c) jittering

Figure 4: Examples of spatiotemporal slices under camera panning, zooming, and jittering.

does the trained prediction coefficient vector. More importantly, subpixel spatial interpolation is implicit in our formation and therefore LSP automatically achieves subpixel accuracy with a spatially-varying interpolation kernel. Such capability of spatially adaptive subpixel interpolation attributes to the excellent prediction accuracy in the cases of nontranslational motion.

4. EXTENSION OF LSP INTO SLOW AND RIGID MOTION

As motion becomes more observable, two issues need to be addressed during the extension of LSP. The first is the LSP support—instead of using a fixed temporal predictor neighborhood in the LSP support as shown in Figure 2, we need to adaptively select it from the motion characteristic observed from the causal past. We will present a frame-based scheme of updating temporal neighbors in LSP (spatial neighbors are kept fixed because temporal coherence is relatively more important than spatial one for video). The second is the motion-related phenomenon such as occlusion, which calls for the tradeoff between space and time. We will demonstrate that LSP automatically achieves the adaptation between spatial and temporal predictions.

4.1. Backward adaptive update of predictor support

The basic requirement is that the support of MV's distribution should be covered by the support of LSP such that the iso-intensity constraint along the motion trajectory can be exploited. Note that adaptive selection of LSP support does not require the segmentation of video, which is often inaccurate and time-consuming. Instead, we target at extracting the information only about the distribution of MVs from video (i.e., what are the dominant motions?). Such reduction significantly simplifies the problem and well matches the coding applications where accurate segmentation is not necessary.

We propose to solve the problem of estimating the distribution of MV under a maximum-likelihood (ML) framework. ML estimation of MV distribution is formulated as follows. Given a pair of video frames, say X, Y, what is the distribution of MV that maximizes the likelihood function, that is, $P(\vec{v} \mid X, Y)$? Note that such problem is different from Bayesian estimation of MV [26]. Our target is not the MV field $\vec{v} = [v_1, v_2]$ but its distribution function because adaptive selection of predictor support only requires the knowledge about dominant MVs.

Let us assume that the image domain Ω can be partitioned into R nonoverlapping regions $\{\Omega_i\}_{i=1}^{R}$ each of which corresponds to an independent moving object with MV of $\vec{v}^i = (v_1^i, v_2^i)$. So theoretically, the likelihood function of MV can be written as

$$P(\vec{v} \mid X, Y) = \sum_{i=1}^{R} r_i \delta(v_1 - v_1^i, v_2 - v_2^i), \tag{4}$$

where $r_i = |\Omega_i|/|\Omega|$ is the percentage of the ith moving object and $\delta(\cdot)$ is the Dirac function. If we inspect the normalized cross-correlation function c_{XY} between X and Y defined by [27]

$$c_{XY}(v_1, v_2) = \frac{\sum_{k_1,k_2} X(k_1,k_2)\, Y(k_1 - v_1, k_2 - v_2)}{\left[\sum_{k_1,k_2} X^2(k_1,k_2) \sum_{k_1,k_2} Y^2(k_1,k_2)\right]^{1/2}}, \tag{5}$$

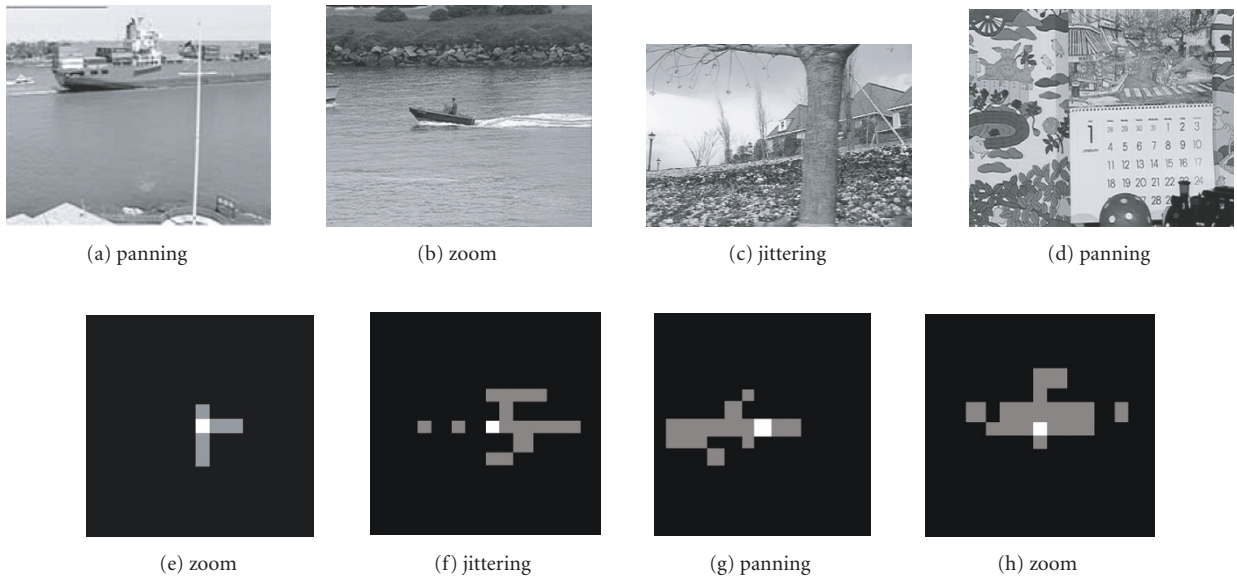
it will have peaks at (v_1^i, v_2^i) [28]. The amplitude of the peak at (v_1^i, v_2^i) is proportional to r_i and disturbed by some random noise (correlation between nonmatched pixels). Since we are only interested in the support of $P(\vec{v} \mid X, Y)$, c_{XY} offers a good approximation in practice.

When there are multiple (say $K > 2$) frames available, we simply calculate the $K-1$ normalized cross-correlation functions for each adjacent pair and then take their average as the likelihood function. For small K values, motion across the frames is coherent; averaging effectively suppresses the noise interference and facilitates peak detection. Due to the computational efficiency of FFT, we have found that such frame-by-frame update of LSP support only requires a small fraction of computation in the overall algorithm.

Figure 5 shows some examples of the final peak detection results (after thresholding the averaged cross-correlation function) for different types of motion. The location of peaks determines the support of temporal prediction neighbors in (1). It can be observed that (1) in the case of slow object motion (e.g., *container*), a small support is sufficient to exploit temporal redundancy; (2) as motion gets faster and more complex (e.g., *mobile*), a larger support is generated by the phase-correlation method. The support is often anisotropic—to capture the horizontal motion of camera panning, the LSP support has to cover more pixels along the horizontal direction than along the vertical one.

4.2. Spatiotemporal adaptation of LSP

One salient feature of LSP is that it achieves a good tradeoff between spatial and temporal predictions. For example,



(a) panning (b) zoom (c) jittering (d) panning

(e) zoom (f) jittering (g) panning (h) zoom

Figure 5: Top: starting frame of test video sequences (*container, coastguard, flower-garden,* and *mobile*); bottom: graphical representation of LSP support at the starting frame (white dot indicates the origin, refer to Figure 2).

occlusions (covered/uncovered regions) represent a class of events that widely exist in video with varying scene depth. When occlusion occurs, covered (uncovered) pixels cannot find the correspondence from previous (or future) frames. Such phenomenon essentially reflects the fundamental trade-off between spatial and temporal redundancies—for pixels in occluded areas, temporal coherence is less reliable than spatial one. However, as long as the local training window contains the data of the same occlusion class, LS method can automatically shift the balance towards spatial prediction (i.e., assign more weights to the spatial neighbors than temporal ones).

To illustrate the space-time adaptation behavior of LS method, we use a typical test sequence *garden*. Two pixels locations are highlighted in Figure 6(a): A is in the occluded area where temporal prediction does not work and B is located in nonoccluded areas. At point A, we have found that LS training assigns dominant weights to spatial neighbors, as shown in Figure 6(b); while at point B, it goes the other way—the dominant prediction coefficient is located in temporal neighborhood, as shown in Figure 6(c). Such contrast illustrates the adaptation of LS training to spatial and temporal coherences. Figure 6(d) displays a binary image in which we use white pixels to indicate where the largest LSP coefficient is located in the temporal neighborhood. It can be observed that spatial coherence dominates temporal coherence mostly around smooth or occluded areas.

5. EXTENSION OF LSP INTO FAST AND NONRIGID MOTION

So far, we are constrained to the class of slow and rigid motion where a fixed training window in spatiotemporal domain is used. To handle video sequences with more generic motion, we propose to extend LSP by adapting the training window in the following two manners.

5.1. Camera panning compensation by adaptive temporal warping

A significant source of fast motion in video is camera panning. A fast panning camera introduces global translational motion to the video, which gives rise to irrelevant data in the training window (refer to the red box in Figure 7(a)). Consequently, the gain of LSP often diminishes due to the inconsistency between training data and the targeted motion trajectory. Note that such difficulty cannot be overcome by increasing the temporal window size since the tunnel carved by the object motion relative to the camera is in the slant position.

One convenient solution to compensate the camera panning is via temporal warping [29]. Under the assumption that the camera panning is approximately along the horizontal direction, the global translational motion can be compensated by horizontally shifting the k_3th frame by $(k_3 - 1)d$ pixels, where d is the camera panning speed (pixels per frame). Figure 7 gives an example of shifting two frames $k_3 = 1, 2$ in the case of $d = 1$. Note that such temporal warping simply relabels the indexes of each frame and does not involve any modification of pixel values. Since warping is a deterministic operation, it can be easily reversed at the decoder (assuming the same d is used) and has no impact on the computational cost.

The camera panning speed can be inferred from the peaks in the phase-correlation domain. Unlike [29] that employs irreversible interpolation techniques to achieve subpixel alignment, we only need to consider integer shifts here

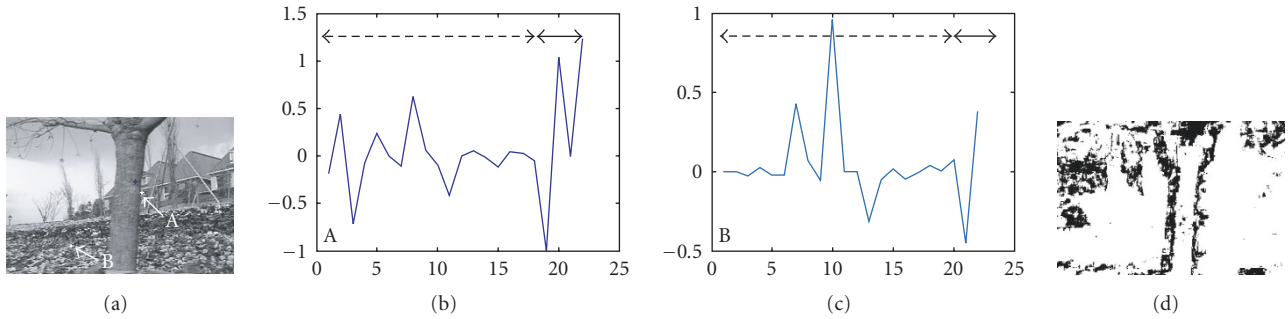


Figure 6: Illustration of space-time adaptation. (a) A and B represent two locations with and without occlusion; (b), (c) LSP coefficient profiles for A and B (dashed and solid denote temporal and spatial neighbors, resp.); (d) a binary image in which white pixels indicate where temporal coherence dominates spatial coherence.

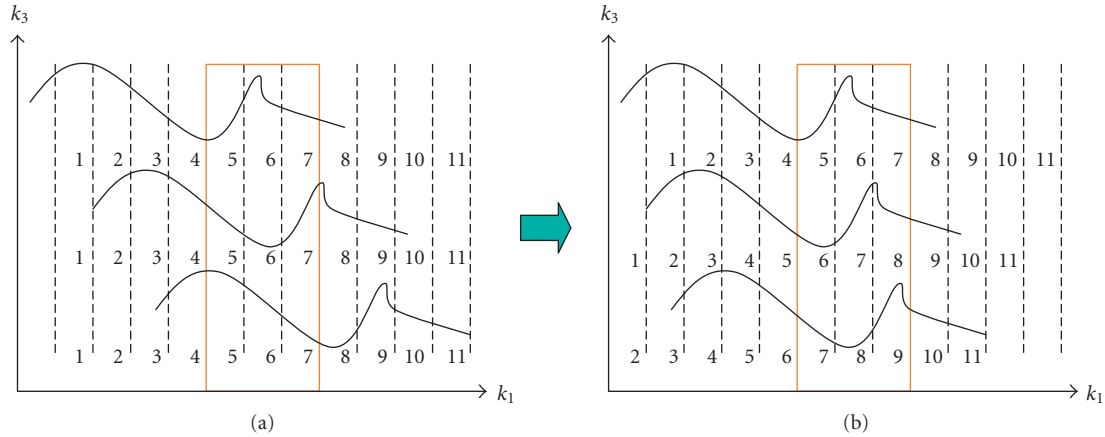


Figure 7: Illustration of temporal warping for camera panning compensation: (a) before compensation; (b) after compensation. Note that more relevant data are located inside the training window (red box) after the compensation.

because LSP itself implements subpixel accuracy interpolation. As shown in Figure 7, the desirable impact of temporal warping is that the fixed spatiotemporal window contains more relevant data suitable for LS training after the compensation of camera panning. The gain brought by such camera panning compensation will be justified later by experimental results (refer to Figure 13).

5.2. Forward adaptation for temporally unpredictable events

In addition to fast camera panning, change of camera panning/zooming speed or disturbance of camera positions also has a subtle impact on the efficiency of LSP. Theoretically, we can adaptively choose the training window $C(T_1, T_2)$ for every pixel to reach the optimal prediction efficiency. However, since an optimal training window necessarily involves local characteristics of motion trajectory (not just the distribution of all MVs), it is difficult to achieve the adaptation without explicit estimation or at least segmentation of the MV field.

One compromised solution is to update the training window on a frame-by-frame basis. For simplicity, we opt to fix the spatial window size $T_1 = 3$ and study the adaptive selection of temporal window size T_2 here. Such simplification is based on the empirical observation that varying T_2 often has a more dramatic impact on the efficiency of LSP than varying T_1. Though the update of T_2 can be done in a similar backward fashion to LSP support, we suggest that forward adaptation is more appropriate here because the overhead is negligible (only one parameter per frame). To select the optimal T_2 for each frame, we suggest the adoption of recursive LS (RLS) [30] as an efficient implementation.

To illustrate the importance of adaptively selecting parameter T_2, we compare two video sequences with similar content (a talking person) but acquired in different environments. The first video is acquired by a fixed camera and the second is captured on a bumping moving vehicle (refer to Figure 4(c)). Figure 8 shows the impact of varying T_2

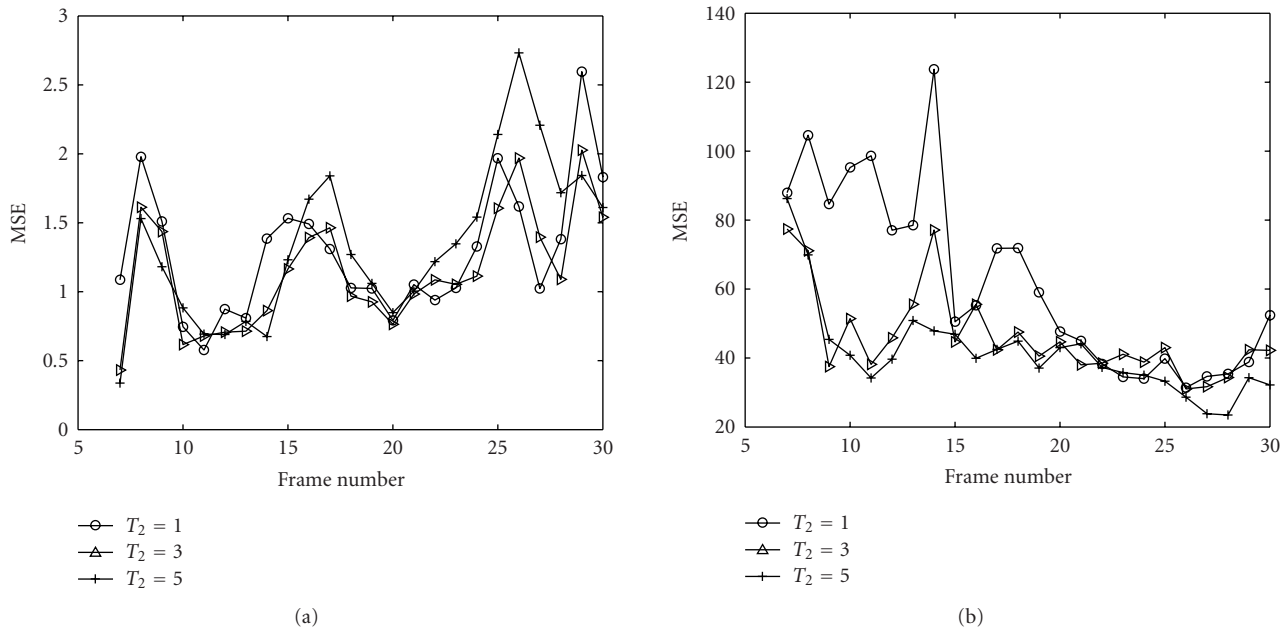


FIGURE 8: Frame-by-frame MSE evolution as a function of T_2 (circle, triangle, and cross correspond to $T_2 = 1, 3, 5$, resp.): (a) *akiyo* sequence—no jittering; (b) *carphone* sequence—with jittering.

(temporal window size) on the efficiency of LSP for two sequences. It can be observed that the optimal T_2 is larger for the second sequence in order to suppress the disturbance of jittering on motion trajectory.

The more challenging situations involve fast and nonrigid object motion that cannot be easily compensated or predicted from the causal past. Note that such events distinguish from occlusions because they are temporally unpredictable (the event of occlusion is at least temporally coherent and occluded pixels can still be predicted from either the past or the future). Fundamentally speaking, such temporally unpredictable events are innovations that do not fit the backward adaptive framework. Therefore, we propose to handle them separately by forward adaption assuming those events are spatially localized. To inform the decoder about the pixels that temporal prediction does not apply, we need to spend a small amount of overhead on coding their boundaries. Therefore, still background and moving objects can be decomposed into different layers [16] and handled by backward LSP and forward MC, respectively.

6. EXPERIMENTAL RESULTS

In this section, we use experimental results to demonstrate the boundary of LSP—for a wide range of video material, LSP is highly effective; in the meantime, we have also found that LSP is inappropriate for certain type of material such as sports video. The MATLAB codes of our implementation are available at http://www.csee.wvu.edu/~xinl/code/LSP.zip.

6.1. Experimental setup

In our implementation of LSP, two issues need to be addressed. First issue is how to select the threshold in determining the LSP support. Due to the variation of phase-correlation function from sequence to sequence, no universal threshold exists. Instead, we suggest an adaptive threshold $th = \max(th_1, th_2)$, where $th_1 = c_{\max}/20$ ($c_{\max}$ is the maximum of c_{XY}) and th_2 is the magnitude of the 12th highest peak in c_{XY}. Second issue is how to handle the degenerated case of LS estimation (i.e., $C^T C$ is not full-ranked). Such situation often occurs in smooth and still background which does not require sophisticated LS optimization; instead, we assign the default equal weights to all coefficients in the prediction support.

Since BMA has been adopted by most existing video coding standards, we use it as the benchmark to show the potential of LSP in video coding. In our implementation of BMA, we choose the parameter setting at the QCIF resolution: full-search, 4×4 block size, search range $[-7, 7]$, quarter-pel accuracy. It should be noted that such setting is similar to the one adopted by H.264 and in favor of prediction accuracy (larger block-size only renders higher residue energy). The overhead of 1584 quarter-pel MVs per frame is often a significant portion especially at low bit rates. Since image borders cause problems to both BMA (e.g., unrestricted MV mode in H.263) and LSP (not enough training samples), we only calculate the MSE for prediction residues ten pixels away from the border. The experimental results are reported for the first 30 frames of all video sequences.

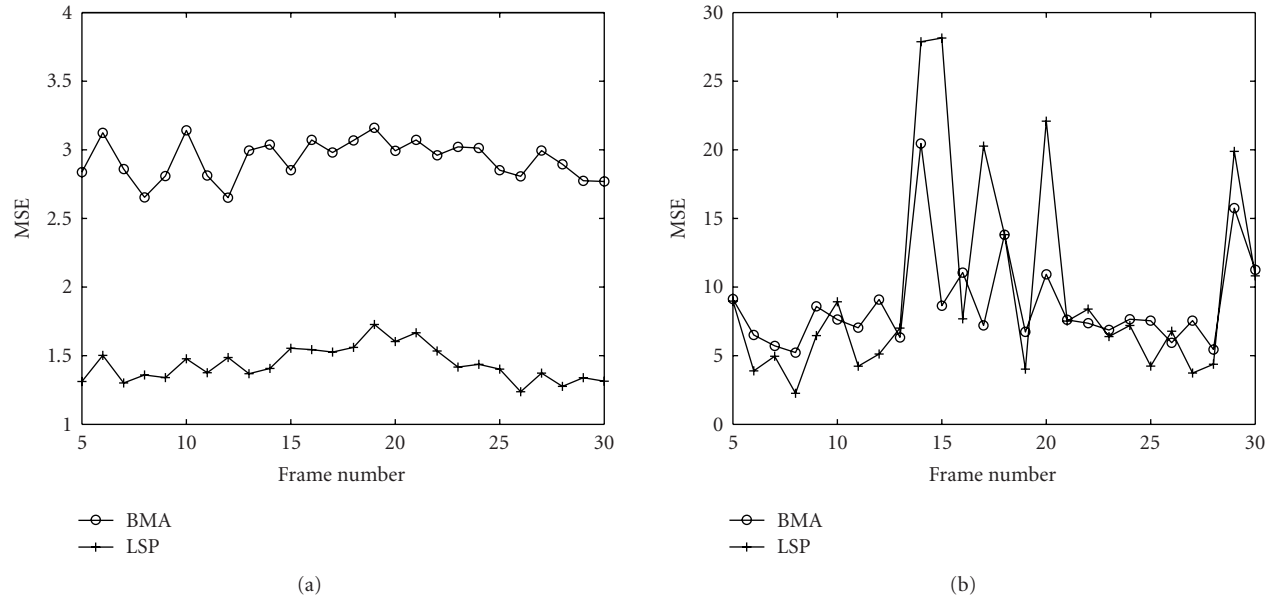


Figure 9: Frame-by-frame MSE comparison between BMA ("∘") and LSP ("+") for sequences with slow translational motion: (a) *container*; (b) *forest*.

6.2. Slow motion

In order to more clearly demonstrate the performance of LSP, we structure the comparison between LSP and BMA into the following three categories with different motion characteristics: (1) slow and translational (e.g., *forest* and *container*); (2) slow camera zoom (e.g., *mobile* and *tempete*); (3) slow nonrigid motion (e.g., *coastguard* and *news*). We believe these three categories of video sequences reasonably cover a wide range of motion in the real world.

Figure 9 shows the frame-to-frame MSE comparison between LSP and BMA for category-1 sequences. When camera is fixed and object moves smoothly (*container*), we observe that the MSE values of both BMA and LSP are small; however, LSP achieves even smaller MSE on the average than BMA (about 3.8 dB reduction). When camera slowly moves (*forest*), uneven camera motion gives rise to peaks in MSE profile of LSP (e.g., frames no. 14, 16, 19 in *forest*). However, the average MSE values between LSP and BMA are still comparable (8.93 versus 8.81); note that the overall coding gain of LSP is still higher than BMA since it does not require any overhead.

The advantage of LSP over BMA becomes even more obvious as slow camera zoom is involved. Figure 10 shows the MSE comparison results for two category-2 sequences.[2] Since block-based model becomes less accurate for zoom-related motion, forward MC suffers from large errors around block boundaries. Especially for the *mobile* sequence containing abundant textures, LSP achieves 1.87 dB gain over quarter-pel BMA (its average MSE is even smaller than that of 1/8-pel BMA) without any overhead. For *tempete* sequence, we note that the large MSE value of frame 27 is due to the rapidly falling feather—a temporally unpredictable event (refer to Figure 11(d)). Therefore, readers need to use extra caution while evaluating the MSE comparison results for this sequence.

Figure 12 compares the MSE results between BMA and LSP for category-3 sequences. When video material contains nonrigid motion such as flowing river or moving body, we observe that forward MC and backward LSP achieve comparable MSE performance though the origins for large errors differ. In forward MC, large MCP errors attribute to the block-based approximation of motion model and the relaxation of iso-intensity constraint due to loss of motion rigidity; in backward LSP, large errors arise from sudden change of motion characteristics. It is interesting to note that for the *news* sequence, backward and forward approaches have complimentary behavior (e.g., valleys in BMA correspond to peaks in LSP). Such observation indicates an improved strategy—switch to forward MC when LSP becomes ineffective (e.g., use the invalid parameter $T_2 = 0$ to indicate the failure of temporal prediction).

6.3. Fast motion

For the category of video material with fast camera panning, we demonstrate how temporal warping improves the prediction efficiency. To simplify the comparison, we take the portion (sized 144×176) of SIF/CIF sequences that does not experience occlusion (it is located on the side opposite to

[2] Since their QCIF versions contain severe aliasing, we use the top-left quarter of CIF sequences in this experiment.

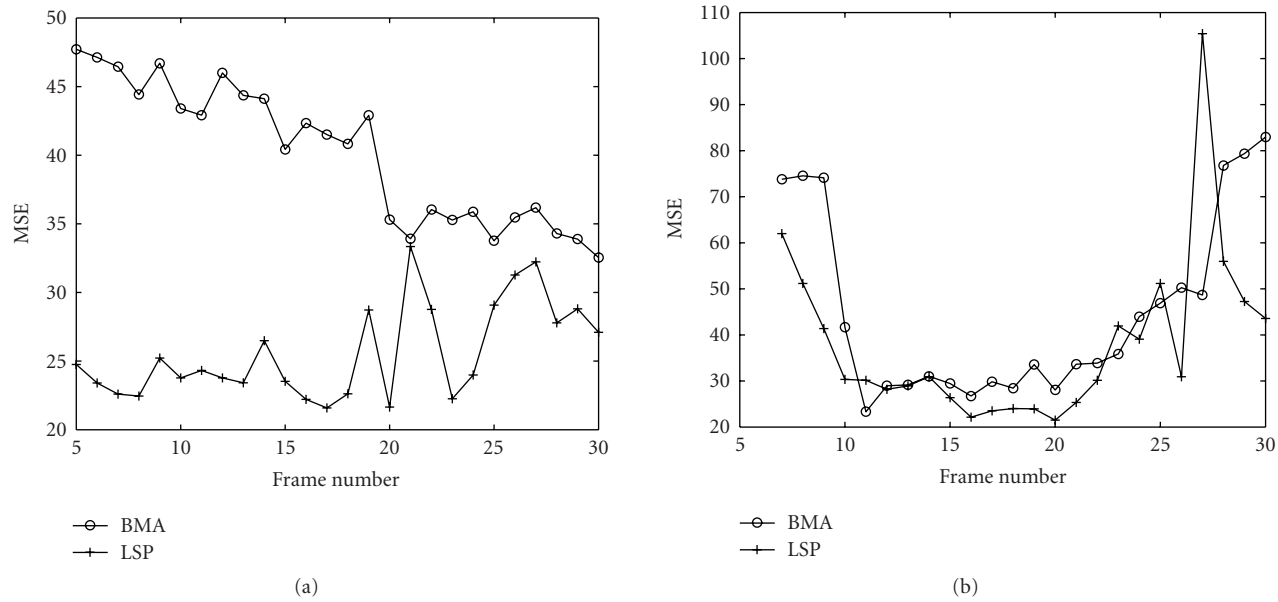


Figure 10: Frame-by-frame MSE comparison between BMA ("∘") and LSP ("+") for sequences with slow zoom motion: (a) *mobile*; (b) *tempete*.

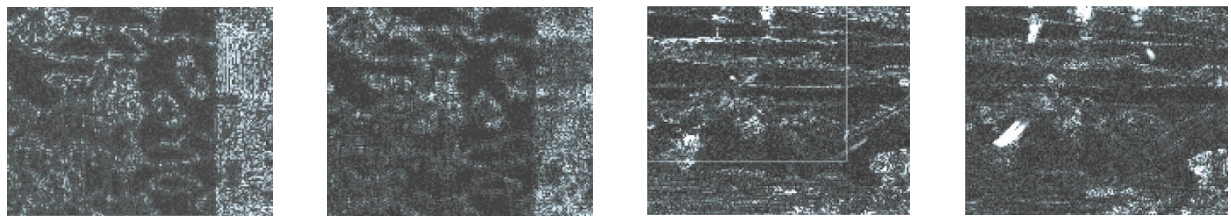


Figure 11: Residue image comparison between BMA and LSP for the 4th frame of *mobile* (a,b) and the 27th frame of *tempete* (c,d): (a) BMA (MSE = 48.0); (b) LSP (MSE = 26.9); (c) BMA (MSE = 48.7); (d) LSP (MSE = 88.6).

the camera panning direction). Figure 13 compares the MSE profiles before and after the compensation with different hypothesized camera panning speeds. As the panning speed d increases, temporal warping gradually straightens the motion trajectory, which renders more relevant data being included to the training window. Thus we observe that the MSE produced by LSP with a fixed spatiotemporal window monotonically decreases with the increasing d.

The last category represents the most challenging situation for LSP, that is, video containing fast nonrigid motion. Such type of video is abundant with temporally unpredictable and spatially localized events, which are not suitable for LSP. Even in forward MC, it often requires the range of motion vectors to be large enough (therefore increased overhead is required). Figure 14 shows the comparison between BMA and LSP for two test sequences *foreman* and *football*. In both sequences, camera is approximately fixed but objects (human head and body) move rapidly and involve deformation. The poor performance of LSP indicates that it has to be combined with forward adaptation as suggested at the end of Section 5.2.

6.4. Computational complexity

The computational bottleneck of LSP is the calculation of covariance matrix C^TC in (3)—it requires $O(N^2M)$ arithmetic operations if implemented straightforwardly [31]. In a typical parameter setting ($T_1 = 3, T_2 = 2, N = 13$), brute force implementation amounts to around 17 K arithmetic operations per pixel. Such prohibitive computational cost is the major disadvantage of LSP (note that encoder and decoder have symmetric complexity since it is backward adaptive). In the literature, there exists fast implementation of calculating covariances by exploiting the overlap of

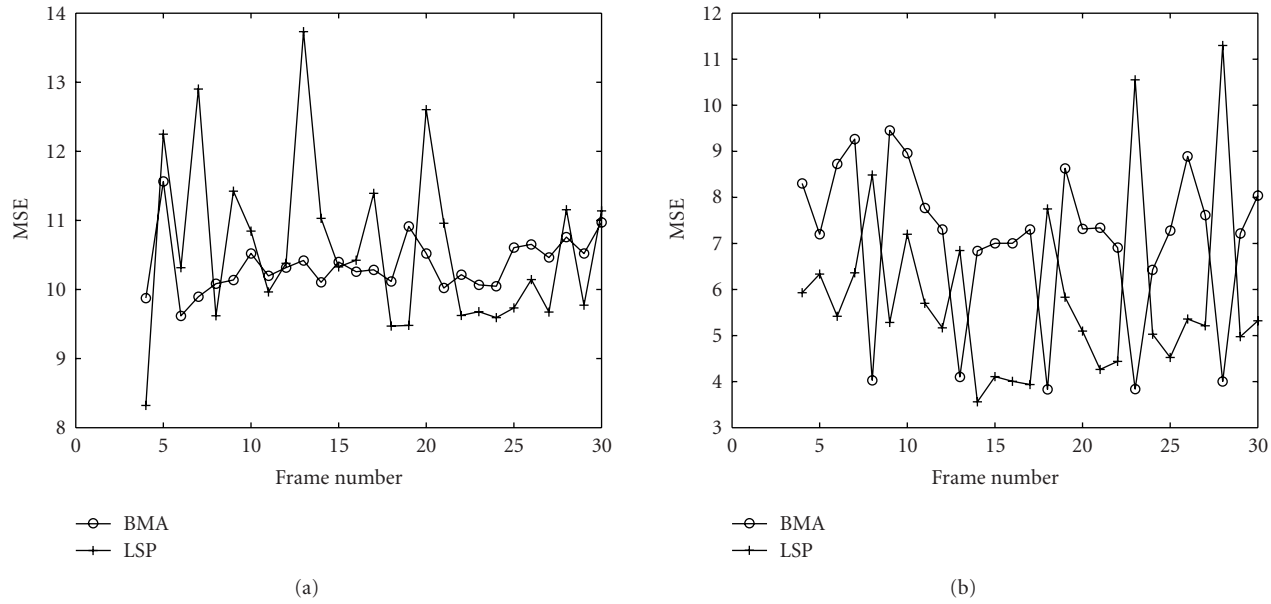


Figure 12: Frame-by-frame MSE comparison between BMA ("∘") and LSP ("+") for sequences with slow zoom motion: (a) *coastguard*; (b) *news*.

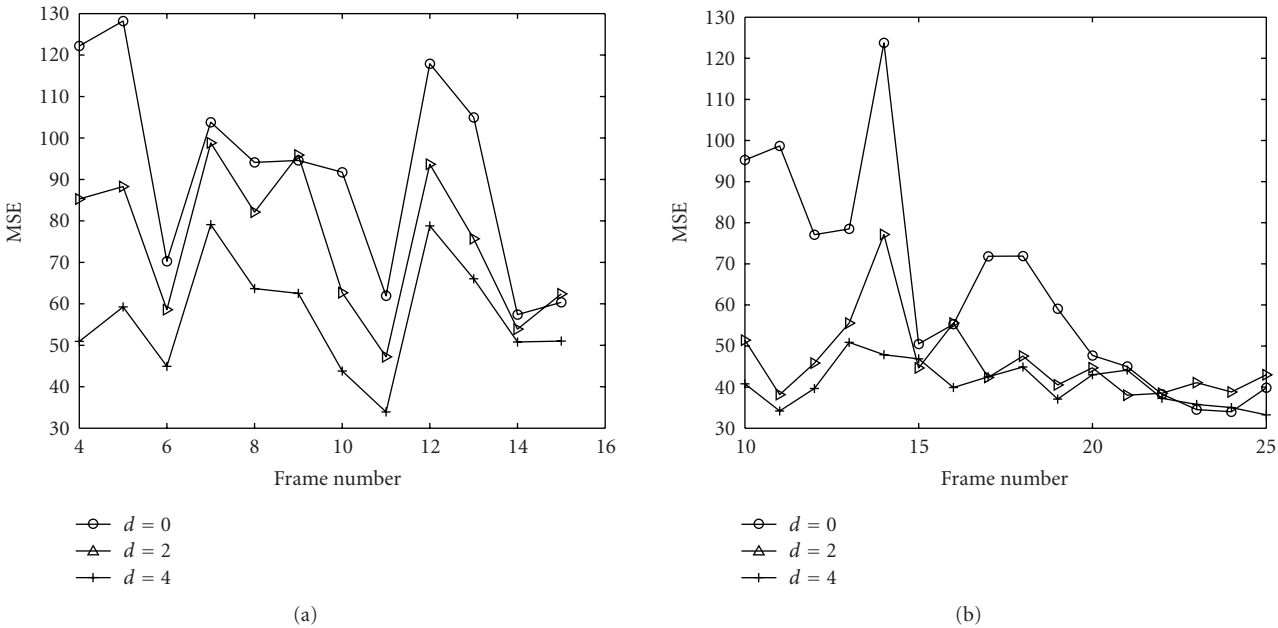


Figure 13: Temporal warping improves the prediction efficiency for video with camera panning: (a) flower-garden and (b) bus.

training window between adjacent pixels. For example, the so-called "inclusion-and-exclusion" technique [32] can effectively reduce the complexity to about 1 K arithmetic operations per pixel. We expect that with fast implementation and more powerful computing resource available, the running time of LSP can be further reduced.

7. CONCLUSIONS AND FUTURE WORKS

In this paper, we challenge the existing paradigm of hybrid MCP coding for video signals by presenting an alternative LS-based backward adaptive predictive coding framework. Motivated by the duality between edge contours in

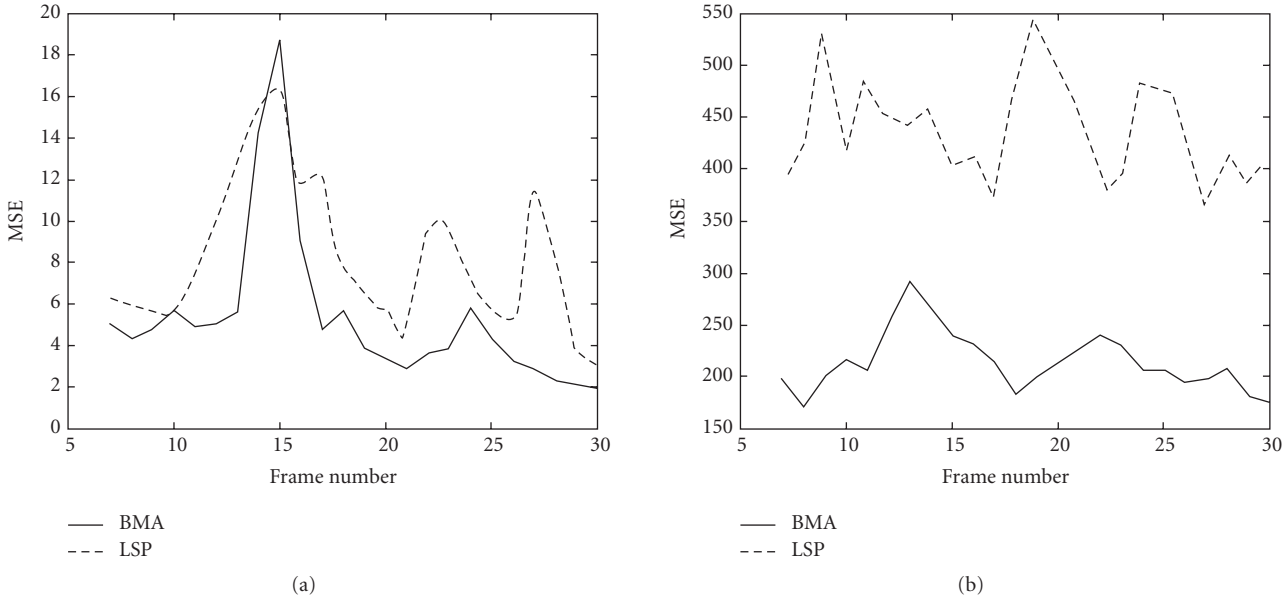


Figure 14: Frame-by-frame MSE comparison between BMA (solid) and LSP (dashed) for sequences with fast nonrigid object motion: (a) slightly-fast *foreman*; (b) ultra-fast *football*.

image and motion trajectories in video, we propose to estimate the instantaneous covariance attributes within a causal spatiotemporal window and use them to derive a linear MMSE predictor. In contrast to explicit ME techniques, ours can be viewed as a localized learning-based approach that implicitly exploits the temporal redundancy. We use experiment results to demonstrate the potential of the proposed backward approach—without sending any overhead, LSP is able to achieve comparable and often smaller MSE values than small block-size, full-search, quarter-pel BMA for a wide range of QCIF test sequences.

There are three directions along which we plan to explore in the future. First, we need to combine backward and forward approaches to more effectively handle the class of video containing fast nonrigid motion. One possible attack is to backwardly segment video to obtain layered representations [16] and adaptively process each layer. Second, we need to design quantization and entropy coding suitable for LSP and study scalability issues under this new framework. Due to backward adaptation, quantization errors could degrade the performance of LSP especially at low bit rates. Third, in order to alleviate the burden of computational demand by LSP on the decoder side, we need to pursue an improved tradeoff between the performance and the cost.

ACKNOWLEDGMENT

The author wants to thank the anonymous reviewers for their insightful comments which were helpful to improve the presentation of this work.

REFERENCES

[1] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Transactions on Communications*, vol. 29, no. 12, pp. 1799–1808, 1981.

[2] R. Srinivasan and K. R. Rao, "Predictive coding based on efficient motion estimation," *IEEE Transactions on Communications*, vol. 33, no. 8, pp. 888–896, 1985.

[3] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[4] G. J. Sullivan and T. Wiegand, "Video compression-from concepts to the H.264/AVC standard," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 18–31, 2005.

[5] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*, Prentice-Hall, Englewood Cliffs, NJ, USA, 2002.

[6] A. Kaup, "Object-based texture coding of moving video in MPEG-4," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 5–15, 1999.

[7] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 41, no. 12, pp. 3445–3462, 1993.

[8] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, 1996.

[9] Z. Xiong, K. Ramchandran, and M. T. Orchard, "Space-frequency quantization for wavelet image coding," *IEEE Transactions on Image Processing*, vol. 6, no. 5, pp. 677–693, 1997.

[10] B. Girod, "Efficiency analysis of multihypothesis motion-compensated prediction for video coding," *IEEE Transactions on Image Processing*, vol. 9, no. 2, pp. 173–183, 2000.

[11] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Transactions on*

Circuits and Systems for Video Technology, vol. 9, no. 1, pp. 70–84, 1999.

[12] M. T. Orchard and G. J. Sullivan, "Overlapped block motion compensation: an estimation-theoretic approach," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 693–699, 1994.

[13] M. T. Orchard, "Predictive motion-field segmentation for image sequence coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, no. 1, pp. 54–70, 1993.

[14] T. Ozcelik and A. K. Katsaggelos, "A hybrid object-oriented very low bit rate video codec," in *Proceedings of 9th Image and Multidimensional Signal Processing (IMDSP '96)*, Belize City, Belize, March 1996.

[15] X. Yang and K. Ramchandran, "Low-complexity region-based video coder using backward morphological motion field segmentation," *IEEE Transactions on Image Processing*, vol. 8, no. 3, pp. 332–345, 1999.

[16] J. Y. A. Wang and E. H. Adelson, "Layered representation for motion analysis," in *Proceedings of IEEE Computer Vision and Pattern Recognition (CVPR '93)*, pp. 361–366, New York, NY, USA, June 1993.

[17] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 604–612, 1993.

[18] D. Brunello, G. Calvagno, G. A. Mian, and R. Rinaldo, "Lossless compression of video using temporal information," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 132–139, 2003.

[19] N. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1984.

[20] X. Li, "On exploiting geometric constraint of image wavelet coefficients," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1378–1387, 2003.

[21] X. Li and M. T. Orchard, "Edge-directed prediction for lossless compression of natural images," *IEEE Transactions on Image Processing*, vol. 10, no. 6, pp. 813–817, 2001.

[22] C.-W. Ngo, T.-C. Pong, H.-J. Zhang, and R. T. Chin, "Motion characterization by temporal slices analysis," in *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR '00)*, vol. 2, pp. 768–773, Hilton Head, SC, USA, June 2000.

[23] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang, "Motion analysis and segmentation through spatio-temporal slices processing," *IEEE Transactions on Image Processing*, vol. 12, no. 3, pp. 341–355, 2003.

[24] A. Tekalp, *Digital Video Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1995.

[25] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[26] J. Konrad and E. Dubois, "Bayesian estimation of motion vector fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 9, pp. 910–927, 1992.

[27] A. Rosenfeld and A. Kak, *Digital Picture Processing*, Academic Press, New York, NY, USA, 1982.

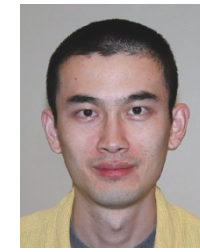
[28] L. G. Brown, "A survey of image registration techniques," *ACM Computing Surveys*, vol. 24, no. 4, pp. 325–376, 1992.

[29] D. Taubman and A. Zakhor, "Multirate 3-D subband coding of video," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 572–588, 1994.

[30] T. Kailath, A. Sayed, and B. Hassibi, *Linear Estimation*, Prentice-Hall, Englewood Cliffs, NJ, USA, 2000.

[31] S. Haykin, *Adaptve Filtering Theory*, Prentice-Hall, Englewood Cliffs, NJ, USA, 4th edition, 2002.

[32] X. Wu, K. U. Barthel, and W. Zhang, "Piecewise 2D autoregression for predictive image coding," in *Proceedings of IEEE International Conference on Image Processing (ICIP '98)*, vol. 3, pp. 901–904, Chicago, Ill, USA, October 1998.



Xin Li received the B.S. degree with highest honors in electronic engineering and information science from the University of Science and Technology of China, Hefei, in 1996, and the Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, in 2000. He was a Member of Technical Staff with Sharp Laboratories of America, Camas, Wash, from August 2000 to December 2002. Since January 2003, he has been a Faculty Member in Lane Department of Computer Science and Electrical Engineering. His research interests include image/video coding and processing. He received the Best Student Paper Award at the Conference of Visual Communications and Image Processing, San Jose, Calif, in January 2001. He is currently serving as an Associate Editor for IEEE Transactions on Circuits and Systems for Video Technology.